# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-263243

(43)Date of publication of application : 11.10.1996

(51)Int.CI. G06F 3/14

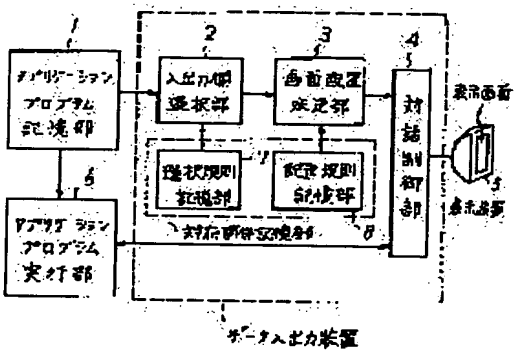
(21)Application number : 07-091911

(22)Date of filing : 27.03.1995

(71)Applicant : TOSHIBA CORP

(72)Inventor : KIMURA YASUHIRO
KANAI TATSUNORI

## (54) METHOD AND DEVICE FOR INPUTTING AND OUTPUTTING DATA AND COMPUTER SYSTEM



(57)Abstract:

PURPOSE: To unnecessitate work for preparing a program for picture display by letting a display picture appear by finding the constitution of a picture corresponding to an application program, and inputting/outputting data between the application program and a user through this appearing display picture.

CONSTITUTION: On the preceding stage of an input/output column selection part 2 at the time of executing the application program, the definition of the type of data to be inputted or outputted by the application program is extracted from the application program. Based on this extracted type of data, the following stage of the input/output column selection part 2 and a picture arrangement deciding part 3 refer to the previously stored relation of correspondence between the type of data and the constitution of the picture to be displayed, store it in an application storage part 1 and find the constitution of the picture corresponding to the application program under execution by an execution part 6. According to this found constitution, the display picture is let appear and data are inputted/outputted between the application program and the user.

CLAIMS

[Claim(s)]

[Claim 1] The display screen which manages the output of the data from [ from a user ] the entry of data to an application program or an application program is made to appear in the display of a computer. In the data I/O approach which outputs and inputs the data between an application program and a user through this display screen The definition of the data type with which an application program performs either [ at least ] an input or an output at the time of application program activation is extracted from this application program. Based on this extracted data type, correspondence relation with the configuration of the screen which should be displayed as the data type memorized beforehand is referred to. Ask for the configuration of the screen corresponding to said application program, and the display screen is made to appear in said display according to this called-for configuration. The data I/O approach characterized by outputting and inputting the data between an application program and a user through this display screen made to appear.

[Claim 2] It is data input output equipment which should be provided to a calculating machine equipped with the indicating equipment which can display the screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program. An extract means to extract the definition of the data type with which this application program performs either [ at least ] an input or an output from an application program, A storage means to memorize the correspondence relation between data type and the configuration of the screen which suited the specification of said calculating machine and which should be displayed about two or more data type, A display-control means to display the display screen on said display according to the correspondence relation memorized by the extract result of said extract means, and said storage means, Data input output equipment characterized by having an input/output control means to perform at least one side of presentation through said display screen of the data outputted from the transmission to said application program of the data inputted through this display screen, and said application program.

[Claim 3] The display means which can display the screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program, An extract means to extract the definition of the data type with which this application program performs either [ at least ] an input or an output from said application program, A storage means to memorize correspondence relation with the configuration of the screen which should be displayed on data type and said display means about two or more data type, A display-control means to display the display screen on said display means according to the correspondence relation memorized by the extract result of said extract means, and said storage means, An input/output control means to perform at least one side of presentation through said display screen of the data outputted from the transmission to said application program of the data inputted through this display screen, and said application program, Computer equipment characterized by having an activation means to perform at least one side of the output of activation of said application program using the data transmitted by this input/output control means, and the activation result of said application program to an input/output control means.

## DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Industrial Application] In case it performs an application program in a computer, this invention makes the display screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program to an application program appear in the display of a computer, and in order to perform the data I/O approach which outputs and inputs the data between an application program and a user through this display screen, and such a data I/O approach, it relates to the data input output equipment which should be beforehand provided to the computer.

[0002]

[Description of the Prior Art] When the program which performs I/O with a user through the screen of terminal units, such as a bit mapped display, and character display or graphic display, was developed, the programmer performed the screen design conventionally bearing in mind a limit (it being called a "specification" below) of the size and the alphabetic character which can be displayed of the screen which the terminal unit (computer equipped with a display) used first has, a graphic form, etc. Next, the program which displays the screen needed to be created combining the control code and command which the windowing system which is operating the terminal unit to be used or on the terminal unit has. Creation of such a program was what that was conscious of arrangement of the alphabetic character on a screen, or a graphic form it is fine and is complicated.

[0003] Furthermore, it was dependent on the model (specification) of specific terminal unit, and the program for the screen display which carried out in this way and was created newly had to create the program, when a new terminal unit was used.

[0004] This means that the program for the screen display from which only the number of models of a terminal unit differs to one kind of application program must be created. Therefore, when the programmer set the number of application programs to n and the number of the models of terminal unit was set to m, only the number of nxm had to create the program for a screen display.

[0005] The software called a form generator or a GUI builder may be used by making creation of the program which outputs and inputs using such a screen into the approach of easy-izing. Such software generates automatically the program which displays the screen, if a programmer defines the arrangement on a screen to use for I/O. Although a part of design of arrangement of a screen became comfort by using such software, since the design of a screen and the design of the body of an application program were divided, when one side was changed, another side also needed to be changed according to it. Moreover, the program which carried out in this way and was created is what was dependent on the model of specific terminal unit too. Therefore, the problem that only the number of nxm was required for creation of the program for a screen display was not solved.

[0006]

[Problem(s) to be Solved by the Invention] This invention aims at enabling it to create the application program which outputs and inputs through a screen in the form independent of the model of terminal unit while it makes creation of a program easy by doing unnecessary the activity which performs the detailed design of a screen for every application in development of the application program which outputs and inputs through a screen in view of the trouble mentioned above, and creates the program for a screen display.

[0007]

[Means for Solving the Problem] This invention (1st invention) makes the display screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program appear in the display of a computer. In the data I/O approach which outputs and inputs the data between an application program and a user through this display screen The definition of the data type with which an application program performs either [ at least ] an input or an output at the time of application program activation is extracted from this application program. Based on this extracted data type, correspondence relation with the configuration of the screen which should be displayed as the data type memorized beforehand is referred to. Ask for the configuration of the screen corresponding to said application program, the display screen is made to appear in said display according to this called-for configuration, and it is characterized by outputting and inputting the data between an application program and a user through this display screen made to appear.

[0008] This invention (2nd invention) is data input output equipment which should be provided to a calculating machine equipped with the indicating equipment which can display the screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program. An extract means to extract the definition of the data type with which this application program performs either [ at least ] an input or an output from an application program, A storage means to memorize the correspondence relation between data type and the configuration of the screen which suited the specification of said calculating machine and which should be displayed about two or more data type, A display-control means to display the display screen on said display according to the correspondence relation memorized by the extract result of said extract means, and said storage means, It is characterized by having an input/output control means to perform at least one side of presentation through said display screen of the data outputted from the transmission to said application program of the data inputted through this display screen, and said application program.

[0009] A display means by which the calculating-machine equipment concerning this invention (3rd invention) can display the screen which manages the output of the data from [ from a user ] the entry of data to an application program, or an application program, An extract means to extract the definition of the data type with which this application program performs either [ at least ] an input or an output from said application program, A storage means to memorize correspondence relation with the configuration of the screen which should be displayed on data type and said display means about two or more data type, A display-control means to display the display screen on said display

means according to the correspondence relation memorized by the extract result of said extract means, and said storage means, An input/output control means to perform at least one side of presentation through said display screen of the data outputted from the transmission to said application program of the data inputted through this display screen, and said application program, It is characterized by having an activation means to perform at least one side of the output of activation of said application program using the data transmitted by this input/output control means, and the activation result of said application program to an input/output control means.

[0010]

[Function] According to this invention, the application program which outputs and inputs data transmits the data type which wants to output and input to data input output equipment. Data input output equipment compounds a screen required for I/O of the data of the received data type, and outputs and inputs the data between an application program and a user using the screen.

[0011] The 1st invention is a thing about the operations sequence performed at the time of activation of an application program. Since such actuation is performed, the 2nd invention is a thing about the means with which the system side of a computer should be equipped. The data input output equipment said here points out the part which realizes the function, when the function equivalent to the equipment which equips the interior of a computer with the means virtually mentioned above by installing software in the system of a computer is formed. The 3rd invention is a thing about the configuration of computer equipment which performs actuation mentioned above.

[0012] In order to create a screen automatically from the data type which wants to output and input at the time of application program activation, the programmer of an application program specifies the fine arrangement on a screen, and it becomes unnecessary thus, to create the program for a screen display beforehand.

[0013] Moreover, since the correspondence relation with the configuration of the screen which should be displayed as the data type memorized in order to use, in case a screen is created automatically suits the model of terminal unit, it becomes possible to use the same application program with the terminal unit with which plurality differs, and it can abolish that an application program is dependent on the model of terminal unit.

[0014] Furthermore, in case it will perform which application program with every terminal unit if this data input output equipment does not need to prepare a different thing for every application program, and sets the number of application programs to n, the number of the models of terminal unit is set to m, and this data entry unit recognizes even m piece existence, I/O of the data through a screen can realize it automatically.

[0015]

[Example] One example of this invention is explained.

[0016] An application program passes the definition of the data type of data which wants to output and input to the data input output equipment of this example. The data type used at this time is possible also for using the data type which the programming language which has described the application program has, and is "The Common Object Request Broker, for example. : Architecture

andSpecification" (Object Management Group issue) The data type independent of specific programming language like CORBA/IDL currently indicated can also be used.

[0017] Here, it is CORBA/IDL. Although the example in the case of using is shown, also when using other data type, treating like this example is possible.

[0018] In order to give the definition of data type to the I/O column selection section 2 (it mentions later) of data input output equipment, it is necessary to express data type as data which can be interpreted by the I/O column selection section 2. This is C as shown in drawing 2 and drawing 3 (continuation of drawing 2 ). It is CORBA/IDL by the structure of language. It is possible to realize with the data type expressing data type. In drawing 2 and drawing 3 typecode The structure with the identifier to say is the data type expressing data type. typecode Inside of the structure kind the field -- int char the structure -- ** -- if it is the structure about information required for each type of definition in the case of a compound die [ like / express the class of said mold and / an array ] whose structure is t_struct ** -- it has in the field to say.

[0019] CORBA/IDL **** -- as the meta data type expressing data type TypeCode The mold to call is defined. It can set to drawing 2 and drawing 3 . typecode The structure of a mold is CORBA/IDL. It can set to data type. TypeCode C It changes into the data type of language.

[0020] Hereafter, the data type definition passed to the I/O column selection section 2 from an application program in this example is CORBA/IDL. It explains as what is passed as data of a TypeCode mold.

[0021] The example of the application program which uses the data input output equipment of this example for drawing 4 and drawing 5 (continuation of drawing 4 ) is shown. The application program of this example is CORBA/IDL. Data type and C It is described with the object-oriented language with the syntax of language.

[0022] It is uim in the illustrated program. It is an object corresponding to the data input output equipment of this invention. uim.input An instruction is the 3rd. It is uim about creating the input column of the data type passed to the argument. It is directing to the object. By this, it is uim. The data type definition of data to input into the I/O column selection section 2 of this example realized by the object is passed. A uim.output instruction is the 3rd similarly. It is uim about creating the output column of the data type passed to the argument. It is directing to the object. uim.event An instruction is an instruction which directs to generate the column which starts the actuation matched with the column like a carbon button. The actuation to start is the 3rd. It gives an argument as a pointer to a function.

[0023] uim.dialog is uim.input. An instruction, a uim.output instruction, and uim.event It uses in order to carry out a hierarchical-relationship definition between an instruction and the I/O column created with a uim.dialog instruction still more nearly recursively. That is, input The 2nd of output, event, and a dialog instruction dialog which serves as parents on the hierarchy can be specified, and a layered structure can be defined as an argument by this. dialog is input. The meaningful settlement on the screen containing output, event, and dialog is shown. uim.input From the created input column, it is get which is an input instruction. A value can be inputted with an instruction.

put which is an output instruction at the output column created by uim.output A value can be outputted with an instruction.

[0024] Moreover, at this example, he is MIT to a display on the screen of the I/O column. Developed X OSF which is a windowing system and its standard toolkit (Open Software Foundation) Motif It is Microsoft although the case where a toolkit is used is explained. Windows of a shrine Also when using, it can carry out by the same method.

[0025] Drawing 1 is an explanatory view explaining the configuration of this invention. In this drawing the application program storage section and 6 1 The application program activation section, 2 the definition of data type to perform I/O from an application program (1) one or more Reception, The close manual power column selection section which chooses the class of I/O column on the display screen required for the I/O corresponding to the received data type, The screen arrangement decision section which opts for the arrangement on the screen of two or more I/O columns which chose 3 by 2, 4 displays on a screen the I/O column chosen by 2 according to the arrangement for which it opted by 3. The dialogue control section which performs the entry of data which displayed the output data passed from the application program (6) on the corresponding output column, and was demanded from the application program (6) through the corresponding input column, and 5 are displays with which the screen containing the I/O column is displayed.

[0026] In addition, with reference to the information memorized by the selection-rule storage section 7 at the time of processing of the I/O column selection section 2, the information memorized by the mapping-rule storage section 8 is referred to in the case of processing of the screen arrangement decision section 3. The storage sections 7 and 8 will have memorized the information on the correspondence relation between data type and the configuration of the screen which suited the specification of this computer and which should be displayed about two or more data type as a whole. The preceding paragraph of the I/O column selection section 2 includes a means to extract the definition of the data type of the data with which an application program performs either [ at least ] an input or an output. The latter part of the I/O column selection section 2 and the screen arrangement decision section 3 moreover, as a whole Based on the aforementioned extract result, the storage section 1 memorizes with reference to the correspondence-related information that the above was memorized, and a means to ask for the configuration of the screen corresponding to the application program currently performed in the activation section 6 is included.

[0027] In addition, when the number of the data with which an application program outputs and inputs is one, even if there is no screen arrangement decision section 3, asking for the configuration of the screen which should be displayed is possible. The dialogue control section 4 includes a means to make a display screen appear according to this called-for configuration, and a means to transmit or to show the data to an application program outputted from an application program through a display screen for the data inputted through this display screen.

[0028] And the above 2-4, and 7 and 8 constitute data input output equipment, and the whole which added 1, 5, and 6 to this constitutes computer equipment.

[0029] The data type definition used as the candidate for I/O is first expressed and indicated by the

application program by TypeCode, respectively. And the I/O column selection section 2 detects the value of this TypeCode from the application program memorized by the storage section 1.

[0030] With reference to the contents of detected TypeCode, according to the selection rule memorized by the selection-rule storage section 7 defined beforehand, the I/O column selection section 2 determines the information on the I/O column corresponding to each data type, and passes it to the screen arrangement decision section 3. The information on the I/O column which the I/O column selection section 2 determines consists of the following elements.

[0031] (1) The configuration of the I/O column displayed.

[0032] It is X about this invention. When carrying out using a windowing system, in order to actually output and input corresponding to the data type specified by an application program, the components which constitute the I/O column called a widget on a screen are displayed. A toggle button is displayed on a widget. ToggleButton A widget and a character string are displayed. Text It displays in the list direction which specified a widget and two or more widgets. There is a RowColumn widget etc. The configuration of the I/O column displayed by the layered structure of a widget as specifically shown in drawing 6 is expressed. For example, the layered structure of the widget created in the program shown in drawing 4 and drawing 5 corresponding to the input instruction of the data type of a "hamburger mold" becomes like drawing 6 .

[0033] (2) Procedure which the generated I/O column needs at the time of activation.

[0034] For example, the I/O column is Scrollbar. It is Scrollbar about the procedure which performs scrolling actuation when the widget is used, and Scrollbar is operated. It is necessary to register. Therefore, "registering into a Scrollbar widget the procedure which performs scrolling actuation" is set to one of the procedure which should be performed after I/O column generation. In the following explanation, the thing of such a procedure that should be performed after I/O column generation will be called "after-treatment procedure."

[0035] (3) Procedure required to read the input data from the user to the column, and pass an application program to the I/O column created with the input instruction from an application program.

[0036] (4) Procedure for displaying on a screen the data received from the application program to the I/O column created to the output instruction from an application program.

[0037] The following and CORBA/IDL The selection rule of the configuration of the I/O column which the I/O column selection section 2 determines corresponding to the main data type, after-treatment procedure, and the procedure over input/output instruction are shown. CORBA/IDL which is not shown here It is possible to apply the same regulation also about data type.

1) Data are boolean. When it is a mold, a ToggleButton widget is used as this type of an I/O column, and it is True and False of truth value by ON of ToggleButton, and OFF. It expresses. Therefore, it becomes one ToggleButton widget like drawing 7 as a configuration of the I/O column. There is especially no after-treatment procedure. Moreover, since the output of data is setting up the condition of ToggleButton appropriately, data output procedure becomes like drawing 9 . Since it is necessary to investigate the condition of ToggleButton similarly for an entry of data, data input

procedure becomes like drawing 10 .

2) Data are short, long, hyper, float, and double. When it is a mold, a Text widget is used as these numerical types of an I/O column. Therefore, the configuration of the I/O column becomes one Text widget like drawing 8 . The after-treatment procedure of the I/O column of a numerical mold is as follows.

- Set it as the need and the sufficient number of alphabetic characters expressing all the values that can treat the number of the maximum alphabetic characters and display digit which can be inputted with the mold. for example, short the range which a mold can treat -32768 to 32767 up to -- it is -- since -- short expressing all the values of a mold -- the need and the sufficient number of alphabetic characters -- 6 it is . Therefore, short It is the number of the maximum alphabetic characters and display digit of a mold which can be inputted in the I/O column 6 It sets up.

- although the mold is expressed, receive only the need and sufficient alphabetic character as a right input. For example, short In the case of a mold, it is 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. - Only 11 characters are received as an input.

[0038] Therefore, there are two as an after-treatment procedure. One is for setting up the width of face of Text appropriately, and it becomes like drawing 11 . in addition, it can set to drawing 11 -- "-- the type of expression -- the need -- as the" approach of asking for several n sufficient alphabetic character -- a mold and several n alphabetic character from -- the becoming table is created beforehand and the approach of lengthening it can be used. Another is for receiving only an appropriate alphabetic character as an input, and when there is an input from a user for that purpose, it needs to perform procedure which checks the validity of an input as shown in drawing 12 . When a Text widget has an input from a user, it is possible to set up so that specific procedure may be performed. Therefore, the second after-treatment procedure becomes like drawing 13 .

[0039] The character string inputted into the last in Text to the input instruction as a procedure over input/output instruction is read, and an application program is passed after changing into a numeric value. The received data are changed into a character string to an output instruction, and the character string is displayed on Text. therefore, data input procedure -- drawing 14 -- like -- Data output procedure becomes like drawing 15 .

3) When data are an enum mold, use RadioBox which consists of two or more ToggleButton widgets and RowColumn widgets as this type of an I/O column. In RadioBox, only one of two or more ToggleButton(s) will be able to be pushed. ToggleButton in the condition of having been pushed expresses the value of data. Therefore, as a configuration of the I/O column, it is RowColumn like drawing 16 . It becomes a configuration of a ToggleButton widget equal to the number of the enumerators which make a widget and it a parent widget.

[0040] Next, in order to use as an I/O column of an enum mold, after-treatment procedure is set up so that generated ToggleButton may operate as above RadioBox(es). However, RowColumn Resource of a widget (parameter which changes a method of operation) In inside, the resource which decides whether to perform automatically a setup required to operate as RadioBox ToggleButton generated as a child widget exists. and Motif **** -- the condition of having set up this resource --

RowColumn The procedure to generate exists. Therefore, RowColumn If this generation procedure is used for generating, especially after-treatment procedure will be lost.

[0041] procedure of as opposed to input/output instruction to the last -- an input instruction -- receiving -- every -- condition (are pushed or not?) of ToggleButton It investigates and the value corresponding to ToggleButton in the condition of being pushed is passed to an application program. Therefore, procedure becomes like drawing 18 . What is necessary is just to set it as the condition of having made ToggleButton corresponding to the value of the received data pushing, to an output instruction on the other hand. Therefore, procedure becomes like drawing 19 R> 9.

4) When data are a string mold, ScrolledText which consists of a ScrolledWindow widget and a Text widget is used for the I/O column of a string mold. Therefore, a configuration turns into a configuration of ScrolledWindow and the child's Text like drawing 17 . In order for the screen of such a configuration to operate as ScrolledText, it is Scrollbar in a ScrolledWindow widget. When operated, it is necessary to carry out a setup which the contents of a display of a Text widget scroll. however, Motif **** -- the procedure which performs automatically a setup required in order to operate as ScrolledText after generating a ScrolledWindow widget and a Text widget is offered. Therefore, if such a procedure will be used for generation of a screen, there will be especially no after-treatment procedure. Procedure which reads the input to the Text widget from a user to an input instruction is performed, and the value is passed to an application program. Procedure which displays the value of the data received to the output instruction on a Text widget is performed. Therefore, the procedure over an output instruction becomes like drawing 21 like drawing 20 in the procedure over an input instruction.

5) As an I/O column of a struct mold when data are a struct mold, it is RowColumn about the I/O column corresponding to each member of struct. What put in into the widget and was summarized to one is used. Therefore, a configuration is RowColumn like drawing 22 . It becomes the form where the widget hierarchy which hits each member is connected downward. The field surrounded with the one-point broken line in drawing 22 is the widget hierarchy of the I/O column corresponding to the mold of each member. This configuration is determined by the procedure as shown in drawing 23 . That is, it is 1 in TypeCode first. It is the configuration for which investigated the information about the mold of the member of eye watch, and opted for the configuration of the I/O column for the mold, and it opted 1 It considers as the I/O column for the members of eye watch. It is RowColumn about the widget tree of each member which repeated this about each member of sturct and was obtained. A widget tree like drawing 22 which ties downward and is obtained is considered as this whole struct type of I/O column configuration.

[0042] Next, although it is after-treatment procedure, this also serves as decision of the configuration of the I/O column, and same procedure. That is, the after-treatment procedure at the time of generating the I/O column of the mold of the member about one member is determined, and this is repeated about each member of a sturct mold. The whole procedure mentioned as a result turns into this struct type of after-treatment procedure. If the above is made into a flow chart, it will become like drawing 24 .

[0043] Although it is the procedure finally performed to the input/output instruction from an application program, to an input instruction, it is 1 first. Input procedure according to the mold of the member is performed about the member of eye watch, and the value of the member is acquired. This is repeated about each member and the value of each member is acquired. Since the value of the original data will be decided if the value of all members is acquired, the value is passed to an application program. The received value is investigated to an output instruction and it is 1. The value of the member of eye watch is acquired. Next, the output procedure according to the mold of the member is performed, and it is 1. The value of the member of eye watch is displayed on a screen. The whole struct mold data of a basis is displayed on a screen by repeating and performing this about all members. Therefore, the procedure over an output instruction becomes like drawing 26 like drawing 25 in the procedure over an input instruction.

6) Data are union. It is union that the data of various molds can be stored corresponding to one storage region each time when it is a mold. It is a mold. It is the union about the candidate of data type who can store in a storage region. It is called the member of a mold. CORBA/IDL union In a mold, the information which identifies the value of current data and the identifier of the member corresponding to the data by which current storing is carried out, i.e., the data type by which current storing is carried out, is saved in a storage region. Therefore, union Since the column for the identifiers of a member and the column for values are needed for the I/O column of a mold, what consists of a field 11 which displays the column RadioBox10 which identifies a member like drawing 27 R> 7, and for values is used. The I/O column according to the mold of the member specified by RadioBox is displayed on the latter. That is, in this field 11, the I/O column corresponding to the mold of each member is generated in the non-display condition, and it is displayed into a field 11, only the I/O column for members corresponding to ToggleButton in the condition of having been pushed in RadioBox being used as a display condition. The widget hierarchy of such a struct blocking output column becomes like drawing 28 . Setting to this drawing, 12 is RowColumn. It is RadioBox which consists of a widget and that child's ToggleButton widget, and 13 is union. It is the widget tree of the I/O column according to the mold of each member. All the widget trees of each I/O column for members make the Form widget the parent widget, and this Form widget is equivalent to the field 11 of drawing 27 . In addition, it is obtained by repeating the procedure of investigating the information about the mold of the member of TypeCode and deciding the I/O column for the mold like the case of a struct mold about the I/O column for each members among the widget trees of drawing 28 , about each member. Therefore, the procedure of determining the whole widget tree of drawing 28 becomes like drawing 29 .

[0044] Next, union There are following two as an after-treatment procedure of a mold.

- union Procedure which performs after-treatment procedure according to the mold of the member about each member.
- Procedure set up so that the I/O column of the member corresponding to the ToggleButton may be displayed on an I/O column viewing area, when ToggleButton in RatioBox is pushed.

[0045] About the former, like the case of a struct mold, the after-treatment procedure at the time of

generating the I/O column of the mold of the member about one member is determined, and it is union about this. It repeats about each member of a mold. The whole procedure mentioned as a result turns into this one-eyed after-treatment procedure. Since it is necessary about the latter to perform a setup which performs procedure as shown in drawing 30 to each ToggleButton in RadioBox when ToggleButton is pushed, the procedure which should be performed becomes like drawing 31 .

[0046] It knows which member the data inputted into the current column will hit by investigating the condition of ToggleButton in RadioBox first to an input instruction as a procedure for finally corresponding to input/output instruction from an application program. Next, input procedure corresponding to the mold of the corresponding member is performed, and the value inputted into the I/O column for members which is in the display condition is read. Since the identifier of a member and the value of data were acquired by the above, it is the original union. The value of mold data becomes settled. Therefore, the value is passed to an application program. On the other hand, to an output instruction, the member identification descriptor in the data received first is investigated, and RadioBox is set as the condition that ToggleButton corresponding to the member was pushed. Then, since the I/O column corresponding to the member will be in a display condition, the output procedure corresponding to the mold of a member is performed, and a value is displayed on the I/O column which is in the display condition. As mentioned above, the procedure over an output instruction becomes like drawing 33 like drawing 32 in the procedure over an input instruction.

7) Data are array. It is array when it is a mold. As an I/O column of a mold, about the same poor thing as plurality is used for the I/O column corresponding to the mold of the element. If only the number of an element will be put in order in that case, when there is much number of an element, un-arranging -- a column has not been settled in a screen -- will arise. Then, an upper limit is prepared in the number which puts the I/O column of an element in order, and it is ScrollBar. It enables it to change the range of the element displayed by using. Therefore, the configuration of the I/O column becomes like drawing 34 .

[0047] By the way, it is array when such an I/O column is used. The number of the column which displays the number of elements and an element that the number of elements exceeds the above-mentioned upper limit stops being in agreement. Therefore, array In a mold, the storage region where the input to data and the screen from a user which were passed with the input instruction from an application program is saved is needed. Moreover, ScrollBar It is necessary to set up so that modification of a display rectangle may be suited and the display of each I/O column for elements may be rewritten, after writing out the input from a user to said storage region, when operated. Therefore, union The following three are raised as an after-treatment procedure of a mold.

- array Procedure which performs after-treatment procedure corresponding to the mold of an element to all the I/O columns for elements.
- Procedure which assigns dynamically said storage region which the generated I/O column uses.
- ScrollBar Since it corresponds to scrolling by actuation, it is ScrollBar. Procedure set up so that

procedure like drawing 35 may be performed, when operated.

[0048] Although it is the procedure for finally corresponding to input/output instruction from an application program, to an input instruction, the value which performs input procedure according to the mold of an element first, and is inputted into the I/O column for elements is read, and the value is written in the location of the element with which said storage region corresponds. After repeating this about all the I/O columns for elements, the value saved in the storage region is passed to an application program. On the other hand to an output instruction, the value received first is written in a storage region. The output procedure according to the mold of an element is performed after that, and the value of the element corresponding to each I/O column for elements is displayed. As mentioned above, in the procedure over an input instruction, as shown in drawing 3636 , the procedure over an output instruction becomes like drawing 37 .

[0049] The I/O column selection section 2 determines the configuration of the I/O column, after-treatment procedure, and the procedure over input/output instruction from detected TypeCode for every data set as the I/O object of an application program according to selection rule (the selection-rule storage section 7 memorizes) which were expressed above. Namely, based on whether the detected data type is equivalent to any of the above 1-7, the information on the I/O column mentioned above is searched for for every data with reference to the selection rule corresponding to data type. And the configuration of the determined I/O column, after-treatment procedure, and the procedure over input/output instruction are passed to the screen arrangement decision section 3.

[0050] In addition, selection rule here are what expressed the correspondence relation between data type and the partial configuration (after-treatment procedure and procedure of further as opposed to input/output instruction) of the screen which suited the specification of this computer, and which should be displayed for every data type. At this example, the specification of a computer is X. Although the example of selection rule in the case of being a windowing system was shown, to a computer with another specification, the regulation over each above-mentioned data type becomes another thing. And each data type with which selection rule are set up is carried out in common for every calculating machine of each specification, if what suited the specification of the calculating machine is memorized, a selection-rule body will not depend for an application program on the specification of a specific calculating machine, but activation of it will be attained in a form as it is by every calculating machine equipped with this data input output equipment.

[0051] The screen arrangement decision section 3 opts for the overall configuration of the I/O column on a screen, and arrangement based on the configuration (partial configuration of a screen) of each I/O column received from the I/O column selection section 2. First, the screen arrangement decision section 3 determines how two or more I/O columns are arranged on a screen, when there are two or more data for I/O. In this example, the arranging method for arranging two or more I/O columns in a single tier perpendicularly as an easy screen arranging method (by or width) is used. This is RowColumn about each I/O column. It is realizable by putting in into a widget. In this case, it is RowColumn about the widget tree of each I/O column like drawing 38 as an overall screen configuration. It becomes the form connected downward.

[0052] The screen arrangement decision section 3 opts for use of the means which enables access to the whole I/O, when it asks for the magnitude of the whole I/O column at the time of arranging two or more I/O columns in a single tier still as mentioned above and the magnitude of this whole I/O column exceeds the magnitude of the screen of a display 5. The whole is specifically put in into ScrolledWindow and it is made to be settled in a screen. For example, when there are too many I/O columns of a screen configuration like drawing 38 and they use ScrolledWindow, the screen configuration becomes as it is shown in drawing 3939 .

[0053] Such an arranging method is realized by referring to the mapping rule memorized by the mapping-rule storage section 8. it is for asking for the configuration of the whole screen based on the partial configuration of two or more screens where mapping rule were searched for for each [ these ] data of every from being for the above-mentioned selection rule showing the partial configuration of the screen which suited the specification of a computer corresponding to each data type, and which should be displayed, and two or more data contained in an actual application program being alike, respectively, receiving, and asking for the partial configuration of a screen.

[0054] In addition, for the mapping rule illustrated here, the specification of a computer is X. Although it is an example in the case of being a windowing system, the computer with another specification memorizes another mapping rule. For example, if the magnitude of the whole I/O column when putting in order and arranging two or more I/O columns in a single tier exceeds the magnitude of a screen in the case of the computer which does not have the scrolling feature of a screen, the I/O column will be assigned to two or more pages, and the whole I/O column will be shown to a user by the page switch function.

[0055] The screen arrangement decision section 3 passes the overall screen configuration of the I/O column determined by doing in this way, and the after-treatment procedure which the I/O column selection section 2 determined and the procedure over input/output instruction to the dialogue control section 4.

[0056] Drawing 40 shows the layered structure of the widget of the screen configuration created in the screen arrangement decision section 3 to the application program of drawing 4 and drawing 5 . Furthermore, drawing 41 shows the example of the screen displayed corresponding to it.

[0057] The dialogue control section 4 is based on the screen configuration of the whole I/O column received from the screen arrangement decision section 3, and is X. The I/O column is displayed on a screen by performing after-treatment procedure as which it drew and the I/O column selection section 2 determined ** after that using the various procedure of a windowing system.

[0058] Moreover, the dialogue control section 4 performs processing to the input/output instruction from an application program. That is, to an input instruction, procedure over the input instruction for which the I/O column selection section 2 opted is performed, the input from a user is read, and an application program is passed. Moreover, the data received by performing procedure over the output instruction for which the I/O column selection section 2 similarly opted to an output instruction are displayed on a screen.

[0059] The actuation at the time of applying the procedure described above to the following data

type is explained.

(1) ** foo, bar, and baz An enum mold called example_enum with three identifiers to say is defined, and it is (2). A struct mold called example_data_type with three members of a short, string, and example_enum mold is defined. Below, the case where this example_data_type type of data are outputted and inputted is explained.

[0060] First, an application program changes the definition of an example_data_type mold into the data of a TypeCode mold, and passes it to the I/O column selection section 2. In addition, although the above-mentioned example explained that the I/O column selection section 2 performed actuation which extracts the definition of the data type with which an application program outputs and inputs from an application program with reference to the application program storage section 1, the application program activation section 6 may perform extract actuation, and may pass an extract result to the I/O column selection section 2. Namely, wherever it may perform the extract of a definition of the data type from an application program by the data-input-output-equipment side, and may carry out in the application program activation section, it may pass an extract result to data input output equipment and the above-mentioned extract means may be in [ in this computer equipment ], it does not deviate from the range of this invention. Based on the received TypeCode mold data, the I/O column selection section 2 determines the following matters.

- The layered structure of the widget which constitutes the I/O column.
- After-treatment procedure which should be performed after widget generation.
- The data input manual continuation which should be performed in order to pass the data inputted into the screen to an application program to the input instruction from an application program.
- The data output manual continuation which should be performed in order to display the data passed from the application program on a screen to the output instruction from an application program.

[0061] The layered structure of a widget is decided first. From received TypeCode, the I/O column selection section 2 gets to know that the data for I/O are a struct mold. Then, the I/O column selection section 2 investigates the mold of each member first, and determines the I/O column for the members. A member called member1 is short. Since it is a mold, a Text widget is used as the I/O column. member2 Since the member to say is a string mold, it uses ScrolledText which consists of a ScrollWindow widget and a Text widget as the I/O column. member3 Since it is the enum mold in which the member to say is an example_enum mold and this example_enum mold has three identifiers, as that I/O column, it is RowColumn. RadioBox which consists of a widget and three ToggleButton widgets is used. If the widget configuration of each I/O column for members is decided,

it is RowColumn about them. Let what was stored into the widget be the I/O column of the whole struct. Therefore, the layered structure of the widget which constitutes the I/O column is determined like drawing 42 .

[0062] Next, the I/O column decision section 2 opts for after-treatment procedure, a data input manual continuation, and a data output manual continuation, and passes them to the screen arrangement decision section 3 in accordance with the layered structure of a widget. The screen arrangement decision section 3 opts for the configuration of the whole screen based on the widget hierarchy structure in this. The screen now used for a display is small, and the whole I/O column displayed based on widget hierarchy structure presupposes that it became the magnitude which is not settled in a screen. In this case, the screen arrangement decision section 3 opts for use of ScrollWindow, in order to enable access to the whole column. Therefore, widget hierarchy structure like drawing 43 as a configuration of the whole screen is determined. And the after-treatment procedure passed from this widget hierarchy structure and the I/O column decision section 2, a data input manual continuation, and a data output manual continuation are passed to the dialogue control section 4.

[0063] The dialogue control section 4 is X. The screen according to the widget hierarchy structure of drawing 43 is generated by calling the various procedure of a windowing system. Although after-treatment procedure is performed after that, there is no after-treatment procedure which should be performed in this case. A screen like drawing 44 is displayed by the above. Moreover, the dialogue control section 4 performs an output instruction manual continuation for the data which may have had the input instruction manual continuation performed to the input instruction from an application program to delivery and an output instruction, and displays data.

[0064]

[Effect of the Invention] As explained above, while making easy development of the application program which outputs and inputs data according to this invention, the dependency to a specific I/O device, a specific windowing system, etc. of an application program which perform data I/O can be abolished.

---

## DESCRIPTION OF DRAWINGS

---

[Brief Description of the Drawings]

[Drawing 1] Drawing showing the whole this example configuration.

[Drawing 2] Drawing showing the example of an expression of a definition of data type.

[Drawing 3] Drawing showing the example of an expression of a definition of data type.

[Drawing 4] Drawing showing the example of the application program which outputs and inputs data.

[Drawing 5] Drawing showing the example of the application program which outputs and inputs data.

[Drawing 6] Drawing showing the example which expressed the configuration of the I/O column

displayed according to the layered structure of a widget.

[Drawing 7] Data are boolean. Drawing showing the example of a configuration of the I/O column in the case of being a mold.

[Drawing 8] Data are short, long, hyper, float, and double. Drawing showing the example of a configuration of the I/O column in the case of being a mold.

[Drawing 9] Data are boolean. Drawing showing the example of procedure over the input instruction in the case of being a mold.

[Drawing 10] Data are boolean. Drawing showing the example of procedure over the output instruction in the case of being a mold.

[Drawing 11] Data are short, long, hyper, float, and double. Drawing showing the example of the 1st after-treatment procedure in the case of being a mold.

[Drawing 12] Data are short, long, hyper, float, and double. Drawing showing the example of the 2nd after-treatment procedure in the case of being a mold.

[Drawing 13] Data are short, long, hyper, float, and double. Drawing showing the example of the 2nd after-treatment procedure in the case of being a mold.

[Drawing 14] Data are short, long, hyper, float, and double. Drawing showing the example of procedure over the input instruction in the case of being a mold.

[Drawing 15] Data are short, long, hyper, float, and double. Drawing showing the example of procedure over the output instruction in the case of being a mold.

[Drawing 16] Drawing showing the example of a configuration of the I/O column in case data are an enum mold.

[Drawing 17] Drawing showing the example of a configuration of the I/O column in case data are a string mold.

[Drawing 18] Drawing showing the example of procedure over an input instruction in case data are an enum mold.

[Drawing 19] Drawing showing the example of procedure over an output instruction in case data are an enum mold.

[Drawing 20] Drawing showing the example of procedure over an input instruction in case data are a string mold.

[Drawing 21] Drawing showing the example of procedure over an output instruction in case data are a string mold.

[Drawing 22] Drawing showing the example of a configuration of the I/O column in case data are a struct mold.

[Drawing 23] Drawing showing the example of the method of opting for the configuration of the I/O column in case data are a struct mold.

[Drawing 24] Drawing showing the example of after-treatment procedure in case data are a struct mold.

[Drawing 25] Drawing showing the example of procedure over an input instruction in case data are a struct mold.

[Drawing 26] Drawing showing the example of procedure over an output instruction in case data are a struct mold.

[Drawing 27] Data are union. Drawing showing the example of a configuration of the I/O column in the case of being a mold.

[Drawing 28] Data are union. Drawing showing the example of a configuration of the I/O column in the case of being a mold.

[Drawing 29] Data are union. Drawing showing the example of the method of opting for the configuration of the I/O column in the case of being a mold.

[Drawing 30] Data are union. Drawing showing the example of the 2nd after-treatment procedure in the case of being a mold.

[Drawing 31] Data are union. Drawing showing the example of the 2nd after-treatment procedure in the case of being a mold.

[Drawing 32] Data are union. Drawing showing the example of procedure over the input instruction in the case of being a mold.

[Drawing 33] Data are union. Drawing showing the example of procedure over the output instruction in the case of being a mold.

[Drawing 34] Data are array. Drawing showing the example of a configuration of the I/O column in the case of being a mold.

[Drawing 35] Data are array. Drawing showing the example of the 3rd after-treatment procedure in the case of being a mold.

[Drawing 36] Data are array. Drawing showing the example of procedure over the input instruction in the case of being a mold.

[Drawing 37] Data are array. Drawing showing the example of procedure over the output instruction in the case of being a mold.

[Drawing 38] Drawing which is determined in the screen arrangement decision section and in which showing the example of an overall screen configuration.

[Drawing 39] Drawing which is determined in the screen arrangement decision section and in which showing the example of an overall screen configuration.

[Drawing 40] Drawing which is determined in the screen arrangement decision section and in which showing the example of an overall screen configuration.

[Drawing 41] Drawing showing the example of the screen displayed corresponding to drawing 40 .

[Drawing 42] Drawing showing the example of the layered structure of the widget showing the configuration of the I/O column.

[Drawing 43] Drawing showing the example of the layered structure of the widget showing the configuration of the whole screen.

[Drawing 44] Drawing showing the example of the screen displayed corresponding to drawing 43 .

[Description of Notations]

1 -- Application program storage section

2 -- I/O column selection section

3 -- Screen arrangement decision section

4 -- Dialogue control section

5 -- Display

6 -- Application program activation section

7 -- Selection-rule storage section

8 -- Mapping-rule storage section